

**JAPANESE PATENT APPLICATION,  
FIRST PUBLICATION No. H6-12095**

INT. CL.<sup>5</sup>: G10L 9/00  
9/14  
9/18

PUBLICATION DATE: January 21, 1994  
1

---

TITLE	Speech Decoding Method
APPLICATION NO.	H4-170890
FILING DATE	June 29, 1992
APPLICANT(S)	NTT CORP.
INVENTOR(S)	Kazunori MANO, Takehiro MORIYA and Hiroto SUDA

---

**ABSTRACT**

[Purpose] In a code-excited linear prediction decoding method, to suppress degradations in the coded speech quality due to transmission errors.

[Constitution] A transmission code from a transmission path is passed through an error correcting circuit 30, speech coding information respectively of a past, current and future frame are stored in a buffer 31, and code error detection information of respectively a past, current and future frame are stored in a buffer 34. An error state flag is set up in an error state managing portion 35 based on the code error detection information in the buffer 34. Based on this error state flag, a linear prediction parameter, power parameter and a pitch period, sound source gain parameter stored in the buffer 31 are restored by interpolation in a parameter decoding portion 32. Furthermore, post-filter parameters are changed according to the error state flag in the speech decoding portion 33.

**CLAIMS**

1. A speech decoding method used for reconstructing speech during reception, by the frame, of an error correction code and speech coding information composed of a linear prediction parameter code, a power parameter code, a pitch period code, a noise

vector code of a noise codebook and a sound source gain parameter code; wherein code error detection information respectively for a past frame, a current frame and a future frame is used when coding speech coding information with respect to the frame to be decoded, and the speech of the current frame is reconstructed and restored depending on an error detection state of each frame.

## DETAILED DESCRIPTION OF THE INVENTION

### Field of Industrial Application

The present invention relates to a speech decoding method applied, for example, to a mobile radio system in which code errors frequently occur on the transmission path, for reconstructing speech with little quality degradation due to code errors from speech coding information and code error detection information processed with code error control technology (error correction/detection coding processing etc.).

### Prior Art

As high-capacity speech coding methods of 8 kbit/s or less for digital mobile radio or speech storage services, code-excited linear prediction coding (CELP) and vector sum-excited linear prediction coding (VSELP) are typical formats. These technologies are respectively described in M. R. Schroeder and B. S. Atal, "Code-Excited Linear Prediction (CELP): High-quality Speech at Very Low Bit Rates", Proc. ICASSP-85, 25.1.1, pp. 937-940 (1985) and I. A. Gerson and M. A. Jasiuk, "Vector Sum Excited Linear Prediction (VSELP) Speech Coding 8 kbps", Proc. ICASSP-90, S9.3, pp. 461-464 (1990).

Here, CELP coding shall be explained. In CELP coding, a linear prediction parameter code, a power parameter code indicating the power of the frame, a pitch period code, a noise code and a sound source gain parameter code are transmitted as a single frame of about 5-50 ms as speech coding information. A CELP decoder is shown in Fig. 1. At the CELP decoder, the decoded linear prediction parameter A is set as the coefficient of the linear prediction synthesis filter 12. Using the excitation vector (residual waveform) of the frame obtained immediately previous (in the past) to then and decoded pitch period code L, a past excitation vector is cut out from the adaptive codebook 10 at that period, and a time sequence vector repeating one frame of this is outputted. The code vector indicated by the inputted noise code C is read from the noise codebook 11 as a vector. These time sequence vectors are respectively weighted in accordance with the gain codes  $G (= g_1, g_2)$  inputted at the multipliers 21, 22, after which they are added in the adder 23, the addition output is supplied to the synthesis filter 12 as an excitation vector, and the power gain P is multiplied at the multiplier 24, thereby obtaining reconstructed speech. Furthermore, the post-filter 13 gives a final

---

speech output with reduced sense of noise due to formant emphasis and pitch emphasis.

In a speech transmission system applied to a field of application in which code errors occur, loss of quality of the transmitted speech due to coding errors is suppressed using error correction coding technology. However, in regions of application such as mobile radio in which burst errors frequently occur, it is not possible to obtain an adequate application effect even if a strong error correction code is used. This is because it is difficult to eliminate all errors even if a strong error correction code is used due to the high density of errors in bursts.

If speech is decoded from coded information with uncorrectable errors remaining, then the decoded speech will be largely distorted. In order to hold down the distortion, an error detecting function of checking whether the code of a frame has been correctly decoded is provided, and when an error has been detected, a process different from the usual decoding method, i.e. waveform restoration, is performed to suppress the influence of code errors as much as possible.

As restoration methods used until now, there are the following two types. The first, for example, as described in M. J. McLaughlin, "Channel coding for digital speech transmission in the Japanese digital cellular system", IEICE Radio Communications Systems Research Society, RCS90-27, pp. 41-45, 1990, is a method wherein the parameters of past frames are repeated when the current frame is a frame having code errors or a method of repeating the parameters and further controlling the gain of the current frame depending on the state of past frames.

The other restoration method, as described in Japanese Patent Application, First Publication No. H4-30200, is a method wherein, when there is an error, the current erroneous frame is interpolated from past correct frames and future correct frames. In this case, an interpolation with little distortion can be obtained if the past correct frames and the future correct frames are close in time.

#### Problems to be Resolved by the Invention

However, restoration from only the past ultimately repeats past speech so as to reduce the power of frames with errors and make the distortion indiscernible, in which case the current error frame parameters are determined from only past information, so that although there is no need for a delay of more than the current frame for restoration, the continuity with the next frame is lost, thus causing quality degradation anew.

Additionally, when the interpolation from past and future correct frames according to Japanese Patent Application, First Publication No. H4-30200 is applied to a transmission path with frequent burst errors, the frame after the current frame is not necessarily a correct frame, and there are cases in which past and future correct frames are largely separated in time, thus causing large distortions due to the interpolation.

---

Additionally, when errors persist in several frames after the current frame, the required delay time must be made that much longer in order to use correct future frame information, thus making it unsuitable for applications such as mobile radio which emphasize the real-time property.

Furthermore, in CELP coding, the adaptive codebook and memory of the synthesis filter depend on the decoded results of past frames, and even if there are no coding errors in the current frame, if there are errors in past frames, distortions will occur in the reconstructed speech. The purpose of the present invention is to offer a speech decoding method which resolves these problems and has little quality loss even on a transmission path with burst errors.

### Means for Resolving the Problems

In the present invention, when a code error is detected in the current frame using the code error detection information and speech coding information respectively of past, current and future frames, during reception, by the frame, of an error correction code and speech coding information to reconstruct speech, the speech is reconstructed by interpolation and restoration of the parameters of the current frame according to code error detection information from past and future code error detection information and speech coding information, and when code errors are not detected in the current frame, if there is a code error detection in any of the past and future frames, the parameters of the current frame are corrected according to the error detection information.

### Functions

Due to the employment of the above structure, not only the code error detection information of the past and current frames, but the code error detection information up to the future frames are used to decode the parameters of the current frame when decoding a parameter group from speech coding information.

### Embodiments

Herebelow, the CELP format shall be taken as an example of a high-capacity speech coding system, and an embodiment applying the invention to the CELP format shall be explained. In the present embodiment, a CELP format decoder shall be described, but application to analogous formats or to systems containing analogous structures is also possible.

Fig. 2 is a block diagram of a speech decoding process in an embodiment of the present invention. In the present embodiment, decoding using information of past, current and future frames up to a single frame shall be described. First, codes transmitted through the transmission path are error correction processed by the frame in the error correction circuit 30. The speech coding information is sent from 30 to the buffer 31, and error detection information to the buffer 34. The buffers 31 and 34 take in information from

past and current frames and also information from at least one frame in the future. The error detection information from the buffer 34 enters the error state management portion 35, and the error states of past, current and future frames is decided,

The speech coding information which is the output of the buffer 31 and error state from the error state management portion 35 are sent to the parameter decoding portion 32, where the parameters of the current frame are decoded and interpolated. The speech is reconstructed in the speech decoder 33 by means of the output from the parameter decoder 32. In the error state management portion 35, as shown in Fig. 3, the flag groups (S2, S1, S0) representing the states of past, current and future frames are updated on the basis of the error detection information until the next frame stored in the buffer 34 of Fig. 2. S2 corresponds to the past frame, S1 to the current frame and S0 to the future frame. The flags S2 and S1 are the contents of flags S1 and S0 in the previous frame during updating of the frames for speech coding which have been shifted. The content of the flag S0 is determined by the error detection information of the input frame from the corresponding buffer 34 in the state flag updating portion 51 and the content of the flag S1 (this corresponds to the flag S0 of the previous frame).

Fig. 4 shows an update table for S0. In Fig. 4, when the input to the buffer 34 equals 0, this indicates that error correction has been completely achieved in that input frame, when the input to the buffer 34 equals 1, this indicates that error correction was not possible, i.e. that an error was detected in the frame. When S1 equals 0, if no errors are detected in the input frame, then S0 is made equal to 0, while if an error is detected, then S0 is made equal to 1. When S1 is in the range of from 1 to a predetermined number  $M - 1$ , if no errors are detected in the input frame then S0 is made equal to 0, while if an error is detected then S0 is made equal to  $S1 + 1$ , and the number of consecutive error frames until the corresponding frame is held as the content of the flag. When S1 is equal to  $M$ , if an error is not detected in the input frame, then S0 is made equal to  $M + 1$ , and if an error is detected, then S0 is made equal to  $M$ . Here,  $M$  is the length of consecutive frames over which burst-type errors can be considered to exist, e.g. 6. When S1 is equal to  $M + 1$ , if no errors are detected in the input frame, then S0 is made equal to 0, and if an error is detected, then S0 is made equal to  $M$ . Therefore, when S0 is equal to  $M + 1$ , the case is such that no errors are detected in the corresponding frame, but since it is after consecutive burst errors, it is not made equal to 0, and is only made equal to 0 when no errors are detected in the subsequent frame.

With regard to shifting of the state flags in this manner, the document by M. J. McLaughlin mentioned among the prior art (IEICE Radio Communications Systems Research Society, RCS90-27, pp. 41-45, 1990) takes the current frame as an input, S0 as the flag of the current frame used in conjunction with the past flags, which greatly differs from the present invention wherein the current frame is made S1 and the future frame is made S0, and the flags of the frame subsequent to the frame to be decoded are used.

Next, using the outputs S (S2, S1, S0) of the error management control portion 35

obtained in this way, decoding of the parameters of the speech coding information is performed in consideration of the error states of the past, current and future frames in the parameter decoding portion 32 of Fig. 2. Fig. 5 shows a flow diagram of a process depending on the error state flags in the parameter decoding portion 32. First, in step 101 of Fig. 5, it is determined whether the error state flag S1 of the current frame is 0, and if 0, then process 1 of step 111 is performed. In process 1, since S1 = 0, S0 and S2 are observed on the basis of parameters decoded based on the speech coding information of the current frame, and if S0 or S2 is not 0, then corrections are made so as to make the parameters connect continuously.

Next, in step 102, when S1 is not equal to 0 and the flag is anywhere from 1 to M, this indicates that an error has been detected in the current frame. Therefore, in this case, errors are contained in the speech coding information of the current frame. Thus, the information of a past frame and a future frame must be used. In step 103, if S0 = 0 or M + 1, then the future frame is correct, so that as the process 2 of step 112, the parameters of the current frame are interpolated from the parameters used for decoding in the past frame and the decoding parameters of the future frame. At this time, the weighting of the interpolation can be changed according to the degree of continuity of errors in the flag of S1. For example, if S1 is close to M, then the parameters of the past frame will be parameters obtained by an extrapolation process from process 3 of step 113 to be described, with burst errors continuing, so that by making the weighting of past parameters small and the weighting of future frames large, it is possible to obtain parameters of the current frame with less distortion. In step 103, if S0 does not equal 0 and S0 does not equal M + 1, then an error has been detected in a future frame. In this case, an extrapolation of parameters used in decoding past frames to the current frame is performed in process 3 of step 113. At this time, depending on the degree of continuity of errors in the S1 flag, it is possible to change the weighting of the extrapolation results.

In step 102, if the result is NO, then S1 = M + 1, and this indicates that the current frame is correct after at least the last M frames have had burst errors. Accordingly, in process 4 of step 114, the parameters decoded based on the speech coding information of the current frame are taken as the base, and since S2 = M, there have been burst errors in the past frame, and S0 is equal to 0 or M, so that the parameters are corrected so as to connect continuously in accordance with the value thereof.

Next, the examples of processes 1-4 of Fig. 5 shall be described in connection with actual parameters. In CELP coding, the speech coding information of each frame is composed of a linear prediction parameter A, a power parameter code P, a pitch period code L and a noise code vector C of a noise codebook, and a gain parameter G of a sound source as shown in Fig. 1. First, an example with the linear prediction parameter code A shall be given. As a linear prediction parameter, a line spectrum pair (LSP) with good linear interpolation properties is used during interpolation. If one frame is taken to be 20 ms, the linear prediction parameters will be such as to actually require a parameter every subframe unit of about 5-10 ms. In that case, the parameters

of the subframes are determined by interpolation from the parameters of the previous frame (past frame) or subsequent frame (future frame). Then, if there are code errors in the previous (or subsequent) frame, a direct interpolation of erroneous parameters will yield a large distortion. If the parameters of the current frame are erroneous, it is possible to interpolate the current frame from the previous and subsequent frames as long as there are no errors in the previous and subsequent parameters.

The LSP parameters which are not interpolated immediately after error correction in the past (time  $n - 1$ ), current (time  $n$ ) and future (time  $n + 1$ ) frames are respectively taken as  $\omega(n - 1)$ ,  $\omega(n)$  and  $\omega(n + 1)$ , and the parameters corresponding to the times  $n - 1$  and  $n$  after interpolation in consideration of the error state flags are represented as  $\Omega(n - 1)$  and  $\Omega(n)$ . Additionally, with respect to a  $t$  in the range of  $a \leq t \leq b$ , the interpolation parameter  $\Omega(t)$  at the time  $t$  when the parameters are  $Xa$  at time  $a$  and  $Xb$  at time  $b$  shall be defined as follows:

$$\Omega(t) = f\_A(t; (a, Xa), (b, Xb))$$

The function  $f\_A$  is a function of interpolating between two points  $(a, Xa)$ ,  $(b, Xb)$ , for outputting an interpolation value for time  $t$ . For example, it may represent a linear interpolation.

In process 1 of step 111 in Fig. 5, the error state flag  $S1$  of the current frame equals 0, and there are no errors in  $\omega(n)$ , therefore the following should be done in accordance with the error states of  $S2$  and  $S0$ . With regard to  $t$  of the current frame, when  $n - 1 \leq t \leq n$ , if  $S2 = 0$ , then  $\Omega(t) = f\_A(t; (n - 1, \Omega(n - 1)), (n, \omega(n)))$ ; if  $S2 = 1, \dots, M$ , then  $\Omega(t) = f\_A(t; (n - 1, \omega(n)), (n, \omega(n)))$ . When  $n \leq t \leq n + 1$ , if  $S0 = 0$ , then  $\Omega(t) = f\_A(t; (n, \omega(n)), (n + 1, \omega(n + 1)))$ ; if  $S0 = 0, 1, \dots, M$ , then  $\Omega(t) = f\_A(t; (n, \omega(n)), (n + 1, \omega(n)))$ .

In process 2 of step 112 in Fig. 5,  $\omega(n)$  is not used because there is an error in the current frame parameter, and since  $S0 = 0$  or  $M + 1$ , the parameters of the time  $t$  in the current frame are interpolated according to  $\Omega(t) = f\_A(t; (n - 1, \Omega(n - 1)), (n + 1, \omega(n + 1)))$  using the parameters  $\Omega(n - 1)$  and  $\omega(n + 1)$ . In the process 3 of step 113,  $S0 = 1, \dots, M$ , so that using the past parameters, the parameters of the time  $t$  in the current frame are interpolated according to  $\Omega(t) = f\_A(t; (n - 1, \Omega(n - 1)), (n + 1, \Omega(n - 1)))$ . In this case, a function which expands the bandwidth of the spectrum of  $\Omega(t)$  depending on the values of  $S2$ ,  $S1$  and  $S0$  can be used to weaken the resonance properties of the LSP parameters in the output, thus having an interpolation which prevents distortion.

In process 4 of step 114 in Fig. 5,  $\omega(n)$  can be used, but the previous frame is a burst error, either  $\Omega(n - 1)$  or a parameter such as to obtain a flat spectrum as  $\Omega0$  is used. This is taken as  $\Omega'$ . At  $t$  in the current frame, when  $n - 1 \leq t \leq n$ ,  $\Omega(t) = f\_A(t; (n - 1, \Omega'), (n, \omega(n)))$ ; and when  $n \leq t \leq n + 1$ , if  $S0 = 0$ , then  $\Omega(t) = f\_A(t; (n, \omega(n)), (n + 1, \omega(n + 1)))$ .

$(n + 1)))$ , and if  $S0 = M$ , then  $\Omega(t) = f\_A(t; (n, \omega(n)), (n + 1, \omega(n)))$ .

Next, the processing of the power parameter codes in Fig. 5 shall be considered. This is basically the same as interpolation of linear predication parameters. In the case of power also, the case wherein power is calculated in units of subframes shall be described. The power parameters which are not interpolated immediately after error correction in the past (time  $n - 1$ ), present (time  $n$ ) and future (time  $n + 1$ ) are respectively taken as  $p(n - 1)$ ,  $p(n)$  and  $p(n + 1)$ , and the parameters corresponding to the times  $n - 1$  and  $n$  after interpolation in consideration of the error state flags are taken as  $P(n - 1)$  and  $P(n)$ .

The interpolation parameter  $P(t)$  at time  $t$  of the power which is  $Xa$  at time  $a$  and  $Xb$  at time  $b$  is defined as follows:

$$P(t) = f\_P(t; (a, Xa), (b, Xb))$$

The function  $f\_P$  represents a function such as linear interpolation between the two points  $(a, Xa)$  and  $(b, Xb)$ , or linear interpolation of  $Xa, Xb$  expressed logarithmically.

In process 1 of step 111 in Fig. 5,  $S1 = 0$ , so there are no errors in  $p(n)$ . Therefore, at  $t$  in the current frame, when  $n - 1 \leq t \leq n$ , if  $S2 = 0$ , then  $P(t) = f\_P(t; (n - 1, P(n - 1)), (n, p(n)))$ ; and if  $S2 = 1, \dots, M$ , then  $P(t) = f\_P(t; (n - 1, p(n)), (n, p(n)))$ . If the resulting  $P(t)$  is at least a standard constant (e.g.  $\alpha \geq 1$ ) times greater than  $P(n - 1)$ , then it is made  $\alpha'P(t)$  ( $0 \leq \alpha' \leq 1$ ) to limit the power, thus making it impossible to discern distortion due to sudden power changes. Additionally, when  $n \leq t \leq n + 1$ , if  $S0 = 0$ , then  $P(t) = f\_P(t; p(n), (n + 1, p(n + 1)))$ , and if  $S0 = 1, \dots, M$ , then  $P(t) = f\_P(t; (n, p(n)), (n + 1, p(n)))$ .

In the process 2 of step 112 in Fig. 5, there is an error in the current frame parameters, so that  $p(n)$  is not used, and since  $S0 = 0$  or  $M + 1$ ,  $P(n - 1)$  and  $p(n + 1)$  are used to interpolate the parameters of the time  $t$  in the current frame according to  $P(t) = f\_P(t; (n - 1, P(n - 1)), (n + 1, p(n + 1)))$ . Here, if the power change is too large such that with respect to a constant threshold value (e.g.  $\alpha (\geq 1)$ ),  $P(t) > \alpha P(n - 1)$ , then the power is substituted by  $\alpha'P(t)$  ( $0 \leq \alpha' \leq 1$ ) to limit the power, thus making it impossible to discern distortion.  $\alpha'$  may be changed according to the continuity number of burst errors of  $S1$ .

In process 3 of step 113,  $S0 = 1, \dots, M$ , so that using the past parameters,  $P(t) = f\_P(t; (n - 1, P(n - 1)), (n + 1, P(n - 1)))$ , and the volume of the error frame is weakened to replace  $P(t)$  with  $\alpha'P(t)$  ( $0 \leq \alpha' \leq 1$ ).  $\alpha'$  may be changed according to the continuity number of burst errors of  $S1$ . For example, if  $S1 = M$ , this means that large errors have occurred in at least  $M$  frames until the current frame, so that  $\alpha'$  is made to equal 0 to form a soundless interval, to keep from outputting distorted speech.



In process 4 of step 114 in Fig. 5,  $p(n)$  is usable, but the previous frame is a burst error frame. Therefore, at  $t$  in the current frame, when  $n - 1 \leq t \leq n$ , thus  $P(t) = f\_P(t; (n - 1, P(n - 1)), (n, p(n)))$ . Then,  $P(t)$  is replaced with  $\alpha'P(t)$  ( $0 \leq \alpha' \leq 1$ ). When  $n \leq t \leq n + 1$ , if  $S0 = 0$ , then  $P(t) = f\_P(t; (n, p(n)), (n + 1, p(n + 1)))$ ; if  $S0 = M$ , then  $P(t) = f\_P(t; (n, p(n)), (n + 1, p(n)))$ . When  $S0 = M$ ,  $P(t)$  is made equal to  $\alpha'P(t)$  ( $0 \leq \alpha' \leq 1$ ).

Next, an example of parameter decoding of a pitch period code  $L$  shall be described. The pitch period code  $L$  is a delay value indicating the pitch delay of the adaptive codebook 10 in Fig. 1. Here, one frame is divided into  $k$  (e.g. 2) subframes, and that number of the value of  $L$  is transmitted as pitch information. The pitch delay is not only the pitch period in CELP, but can also have a jumping value of double periods or triple periods. Therefore, in the present embodiment, a case wherein  $k = 2$  will be explained in connection with a method wherein the value of the pitch period between the previous frame and subsequent frame is substituted when erroneous in the current frame. Of course, when the variations in the pitch period are small, and the pitch period of the previous and subsequent frames can be interpolated, it can be determined by interpolation.

The pitch delays immediately after error correction in a past (time  $n - 1$ ), current (time  $n$ ) and future frame (time  $n + 1$ ) frame are respectively taken as  $lk(n - 1)$ ,  $lk(n)$ ,  $lk(n + 1)$ , ( $k = 1, 2$ ), and the parameters corresponding to the times  $n - 1$  and  $n$  after considering error state flags are made  $Lk(n - 1)$ ,  $Lk(n)$ , ( $k = 1, 2$ ). In process 1 of step 111 in Fig. 5,  $S1 = 0$ , so that there are no errors in  $lk(n)$ , and  $Lk(n) = lk(n)$ , ( $k = 1, 2$ ).

In process 2 of step 112 in Fig. 5, there is an error in the parameters of the current frame, so that  $lk(n)$  is not used, and  $L1(n) = L2(n - 1)$ ,  $L2(n) = l1(n + 1)$ . In process 3 of step 113,  $S0 = 1, \dots, M$ , so using past parameters,  $Lk(n) = L2(n - 1)$ , ( $k = 1, 2$ ).

In process 4 of step 114, the current frame is correct, so that as in process 1,  $Lk(n) = lk(n)$ , ( $k = 1, 2$ ). Next, a decoding example for the sound source gain parameter  $G$  shall be described. In Fig. 1,  $G$  is a general term of the group of gains  $g1, g2$  multiplied to the respective sound source signals selected in the adaptive codebook 10 and noise codebook 11. With regard to this  $G$ , parameter decoding is performed in a manner similar to the pitch delay. The sound source gains immediately after error correction in the past (time  $n - 1$ ), current (time  $n$ ) and future (time  $n + 1$ ) frames are respectively taken as  $\sigma k(n - 1)$ ,  $\sigma k(n)$ ,  $\sigma k(n + 1)$ , ( $k = 1, 2$ ), and the parameters corresponding to the times  $n - 1$  and  $n$  after considering the error state flags are taken as  $Gk(n - 1)$ ,  $Gk(n)$ , ( $k = 1, 2$ ).

In process 1 of step 111 in Fig. 5,  $S1 = 0$  and there is no error in  $\sigma k(n)$ , so  $Gk(n) = \sigma k(n)$ , ( $k = 1, 2$ ). In process 2 of step 112 in Fig. 5, there are errors in the current frame parameters, so  $\sigma k(n)$  is not used, and  $G1(n) = G2(n - 1)$ ,  $G2(n) = \sigma 1(n + 1)$ .

In process 3 of step 113,  $S0 = 1, \dots, M$ , so using the past parameters,  $Gk(n) = G2(n - 1)$ ,

( $k = 1, 2$ ). In process 4 of step 114, the current frame is correct, so as in process 1,  $Gk(n) = \sigma k(n)$ , ( $k = 1, 2$ ).

Concerning the decoding of the noise vector code C of the noise codebook, there is not much correlation between frames, so that in the present embodiment, the codes sent to the current frame are directly used for reproduction without regard to the error state flags. In this way, in the parameter decoding portion 32 of Fig. 2, the parameters of the speech coding information are decoded and restored based on the past, current and future error state flags. The result is turned over to the speech decoder 33 to recreate the speech.

The distortion due to transmission errors can be reduced in the speech decoder 33 as well by using past, current and future frame error state flags S (= S2, S1, S0) from the error state managing portion 35 to control the reconstructed speech of the decoded speech, thereby reducing distortion due to transmission errors. That is, in the post-filter 13 of Fig. 1, the coefficient of the post-filter is controlled by using S to suppress auditory distortions. Normally, the post-filter is composed of H(z) as follows.

$$H(z) = \frac{(1 - \eta z^{-1})A(z/\gamma 1)}{((1 - \delta B(z))A(z/\gamma 2))}$$

$$A(z) = \sum_{i=0}^n \alpha_i z^{-i}$$

$$B(z) = \sum_{i=-1}^1 \beta_i z^{-i-\tau}$$

Here,  $\alpha_i$  is the linear prediction coefficient for each frame,  $\beta_i$  is the long-term prediction coefficient and  $\tau$  is the pitch delay. These are determined adaptively from the transmission parameters or decoded speech. Formant emphasis is performed with  $A(z/\gamma 1)/A(z/\gamma 2)$ , pitch emphasis is performed with  $1/(1 - \delta B(z))$ , and a spectral slant correction effect is provided by  $1 - \eta z^{-1}$ . In the present embodiment, H(z) is adaptively varied in each frame with  $\eta$ ,  $\delta$ ,  $\gamma 1$  and  $\gamma 2$  as functions of the error state flags S(= S2, S1, S0). When S2, S1 and S0 are all either 0 or M + 1, normal values are used, but when any of S2, S1 or S0 is not 0 or M + 1 and there is an error frame, there is a high probability of errors being contained in the adaptive codebook and noise code vector, where the noisiness of the output speech can be suppressed by reducing  $\eta$  and  $\delta$ , weakening the pitch emphasis of the sound source and high-region spectral slant, and increasing the degree of formant emphasis by adjusting  $\gamma 1$  and  $\gamma 2$ . Here, by using error state flags up to the future frame of S0, it is possible to continuously perform switching of  $\eta$ ,  $\delta$ ,  $\gamma 1$  and  $\gamma 2$  better than with only S2 and S1, thus enabling the suppression of degradations due to sound quality changes caused by switching between

frames.

### Effects of the Invention

As described above, according to the present invention, it is possible to suppress unpleasant sonic distortions which are superimposed on the decoded speech when code errors from the transmission path remain on the speech coding bits.

### BRIEF DESCRIPTION OF THE DRAWINGS

- Fig. 1 A block diagram of a CELP speech decoder.
- Fig. 2 A block diagram of a speech decoding process according to an embodiment of the present invention.
- Fig. 3 An enlarged view of the error state managing portion 5 of Fig. 2.
- Fig. 4 An updating table for the S0 flag in the state flag updating portion 51 of Fig. 3.
- Fig. 5 A schematic diagram of the processing flow depending on the error state flags.

### Description of the Reference Numbers

10	adaptive codebook
11	noise codebook
12	linear prediction synthesis filter
13	post-filter
21, 22, 24	multiplier
23	adder
30	error correction circuit
31, 34	buffer
32	parameter decoding portion
33	speech decoder
35	error state managing portion

(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号

特開平 6-12095

(43) 公開日 平成 6 年 (1994) 1 月 21 日

(51) Int. Cl. <sup>5</sup>	識別記号	庁内整理番号	F I	技術表示箇所
G10L 9/00	M	8946-5H		
9/14	J	8946-5H		
	G	8946-5H		
9/18	E	8946-5H		

審査請求 未請求 請求項の数 1 (全 10 頁)

(21) 出願番号 特願平 4-170890

(22) 出願日 平成 4 年 (1992) 6 月 29 日

(71) 出願人 000004226

日本電信電話株式会社

東京都千代田区内幸町一丁目 1 番 6 号

(72) 発明者 間野 一則

東京都千代田区内幸町 1 丁目 1 番 6 号日本  
電信電話株式会社内

(72) 発明者 守谷 健弘

東京都千代田区内幸町 1 丁目 1 番 6 号日本  
電信電話株式会社内

(72) 発明者 須田 博人

東京都千代田区内幸町 1 丁目 1 番 6 号日本  
電信電話株式会社内

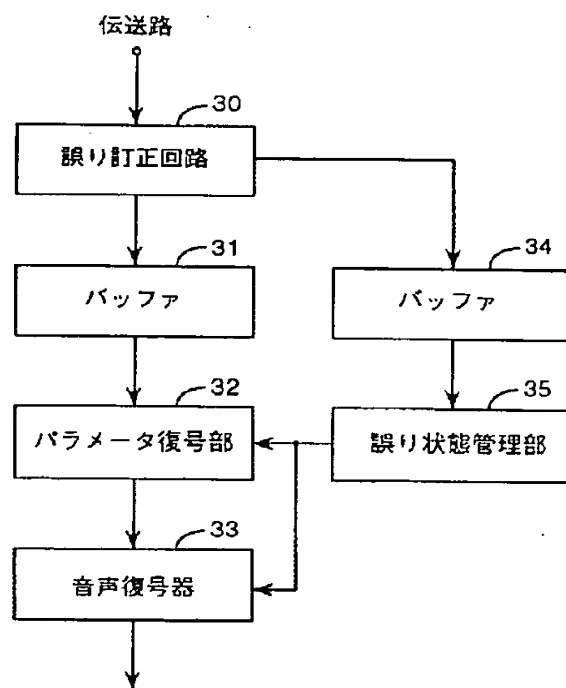
(74) 代理人 弁理士 澤井 敬史

(54) 【発明の名称】 音声復号化方法

(57) 【要約】

【目的】 符号励振線形予測符号化の復号方法において、伝送誤りによる復号音声品質の劣化を抑える。

【構成】 伝送路からの伝送符号を誤り訂正回路 30 とおし、バッファ 31 に過去、現在、及び将来のフレームのそれぞれの音声符号化情報を蓄積し、バッファ 34 に過去、現在、将来のフレームのそれぞれの符号誤り検出情報を蓄積する。バッファ 34 の符号誤り検出情報をもとに誤り状態管理部 35 で誤り状態フラグを設定する。この誤り状態フラグに基づいて、パラメータ復号部 32 において、バッファ 31 に蓄積されている線形予測パラメータとパワパラメータとピッチ周期、音源ゲインパラメータを補間修復する。さらに、音声復号部 33 において、ポストフィルタのパラメータを誤り状態フラグによって変化させる。



## 【特許請求の範囲】

【請求項1】 線形予測パラメータ符号とパワパラメータ符号とピッチ周期符号と雑音符号帳の雑音ベクトル符号と音源ゲインパラメータ符号からなる音声符号化情報と誤り訂正符号とをフレームごとに受信して音声を再生する際の音声復号化方法において、復号しようとするフレームについての音声符号化情報を復号する際、過去のフレーム、現在のフレーム、及び将来のフレームのそれぞれの符号誤り検出情報を用い、各フレーム誤り検出状態に応じて現在のフレームの音声を再生修復する音声復号化方法。

## 【発明の詳細な説明】

## 【0001】

【産業上の利用分野】 この発明は、例えば伝送路上で符号誤りが頻発する移動無線システムへ適用され、符号誤り制御技術（誤り訂正／検出符号化処理等）が施された音声符号化情報と符号誤り検出情報から、符号誤りによる品質劣化の少ない音声を再生するための音声復号化方法に関する。

## 【0002】

【従来の技術】 デジタル移動無線や音声蓄積サービスのための8kbit/s 以下の高効率音声符号化方法としては、符号励振線形予測符号化（Code-Excited Linear Prediction Coding : CELP）、ベクトル和励振線形予測符号化（Vector Sum Excited Linear Prediction Coding : VSELP）といった手法が代表的である。それぞれの技術については、M. R. Schroeder and B. S. Atal : "Code-Excited Linear Prediction (CELP) : High-quality Speech at Very Low Bit Rates", Proc. ICASSP-85, 25. 1. 1, pp. 937-940, (1985年)、および、I. A. Gerson and M. A. Jasiuk : "Vector Sum Excited Linear Prediction (VSELP) Speech Coding at 8kbps", Proc. ICASSP-90, S9. 3, pp. 461-464, (1990年)に述べられている。

【0003】 ここでは、CELP符号化をとりあげて説明する。CELP符号化では、音声符号化情報として、線形予測パラメータ符号と、そのフレームのパワを示すパワパラメータ符号とピッチ周期符号と雑音符号と音源のゲインパラメータ符号を、5～50ms程度を1フレームとして送信する。CELP復号器を図1に示す。CELP復号器では、復号された線形予測パラメータAを線形予測合成フィルタ12の係数として設定する。それまでに得られた直前（過去）のフレームの励振ベクトル（残差波形）と、復号されたピッチ周期符号Lを用いて適応符号帳10からその周期で、過去の励振ベクトルを切り出し、これをフレーム分繰り返した時系列ベクトルが出力され、また、入力された雑音符号Cが示す符号ベクトルが雑音符号帳11からベクトルとして読みだされる。これら時系列ベクトルは、乗算器21、22で入力されたゲイン符号G(=g1, g2)に応じて、それぞれ重み付けされた後、加算器23で加算さ

へ供給され、パワゲインPを乗算器24で乗じて、再生音声を得られる。さらに、ポストフィルタ13により、ホルマント強調、ピッチ強調により雑音感を減らした最終的な音声出力を得る。

【0004】 符号誤りの発生する応用分野に適用される音声伝送方式では、誤り訂正符号化技術を用いて符号誤りによる伝送音声の品質劣化を押さえている。しかし、バースト誤りが頻発する移動無線のような適用領域では、強力な誤り訂正符号を用いたとしても十分な適用効果が得られない。これは、バースト内での誤りの密度が高いため、たとえ強力な誤り訂正符号を用いてもすべての誤りを除去することが難しいことに起因する。

【0005】 訂正しきれない誤りを残したままの符号化情報から音声を復号すると、復号音声は大きく歪む。歪みを小さく押さえるため、誤り検出情報として、フレームの符号が正しく復号できたかどうかをチェックする誤り検出機能を持たせておき、誤りが検出された場合には、通常の復号法とは異なる処理、すなわち、波形修復処理を行ない、符号誤りの影響を極力押さえる工夫がされている。

【0006】 これまでの修復方法としては、次の2種類がある。一つは、例えば、M. J. McLaughlin: "Channel coding for digital speech transmission in the Japanese digital cellular system", 電子情報通信学会 無線通信システム研究会, RCS90-27, pp. 41-45, 1990年、に示すように、現在のフレームが符号誤りのあるフレームの場合に、過去のフレームのパラメータを繰り返す方法や、パラメータを繰り返してさらに、過去のフレームの状態に応じて現フレームの利得を制御する方法である。

【0007】 もう一つの修復方法は、特開平4-30200「音声復号化方式」に、示されているように、誤りがあった場合に、過去の正しいフレームと将来の正しいフレームから、現在の誤ったフレームを補間する方法がある。この場合には、過去の正しいフレームと将来の正しいフレームが時間的に近ければ、歪みの小さい補間を行なうことができる。

## 【0008】

【発明が解決しようとする課題】 しかし、過去のみからの修復は、結局、過去の音声を繰り返して、誤りのあるフレームのパワを小さくして歪みを聞こえにくくするものであり、この場合には、過去の情報のみから現在の誤りフレームパラメータを決定するので、修復のために現フレーム以上の遅延は必要ないが、次のフレームとの連続性が悪くなり不連続性が新たな品質劣化を招くという欠点がある。

【0009】 また、特開平4-30200 による過去と将来の正しいフレームからの補間は、バースト誤りの頻発する伝送路に適用した場合には、現在のフレームの次のフレームが正しいフレームであるとは限らず、過去と将来の

り、補間によってかえって大きな歪みを生じてしまう欠点がある。また、現在フレーム以降何フレームも誤りが続く場合、将来の正しいフレーム情報を使用するためには、その分だけ大きな遅延時間が必要となり、移動無線のように実時間性を重視する用途には適さない。

【0010】さらに、CELP符号化では、適応符号帳や合成フィルタのメモリが過去のフレームの復号結果に依存するので、現在フレームで符号誤りがなくとも過去のフレームで誤りがあれば、再生音声に歪みが生じてしまうという問題がある。本発明の目的は、このような問題点を解決し、バースト誤りのある伝送路でも品質劣化の少ない音声復号化方法を提供することにある。

【0011】

【課題を解決するための手段】本発明では、音声符号化情報と誤り訂正符号とをフレームごとに受信して音声を再生する際、過去、現在、及び将来のフレームのそれぞれの音声符号化情報と符号誤り検出情報を用いて、現在フレームで符号誤りが検出された場合には、過去と将来の符号誤り検出情報と音声符号化情報からその符号誤り検出情報に応じて現在フレームのパラメータを補間修復して音声を再生し、現在フレームで符号誤りが検出されない場合には過去と将来フレームのいずれかに符号誤り検出がある場合には、その誤り検出情報に応じて、現在フレームのパラメータを補正するようにした。

【0012】

【作用】上記構成を採用することにより、音声符号化情報からパラメータ群を復号する際、過去と現在のフレームの符号誤り検出情報だけではなく、将来のフレームまでの符号誤り検出情報を用いて現在フレームのパラメータを復号する。

【0013】

【実施例】以下では、高能率音声符号化方式としてCELP方式をとりあげ、CELP方式へのこの発明の実施例を説明する。本実施例では、CELP方式の復号器について説明するが、これと類似の方式または、類似の構成を含む方式に対しても適用が可能である。

【0014】図2は、本発明の実施例における音声復号処理のブロック図を示す。本実施例では、過去、現在フレームと将来の1フレームまでの情報を用いた復号化について示す。まず、伝送路より送られた符号をフレーム毎に誤り訂正回路30において誤り訂正処理を行なう。30より音声符号化情報をバッファ31に送り、誤り検出情報をバッファ34に送る。31、34のバッファには、過去および現在フレームとさらに少なくとも1フレーム先の情報まで取り込む。バッファ34からの誤り検出情報は、誤り状態管理部35に入り、過去、現在、将来フレームの誤り状態が決定される。

【0015】バッファ31の出力である音声符号化情報と誤り状態管理部35からの誤り状態をパラメータ復号部32

パラメータ復号部32の出力により、音声復号器33で音声を再生する。誤り状態管理部35では、図3に示すように、図2のバッファ34に蓄えられている次のフレームまでの誤り検出情報をもとに、過去、現在、将来のフレームの状態を表すフラグ群352(S2, S1, S0)を更新する。S2が過去のフレーム、S1が現在フレーム、S0が将来フレームに対応する。フラグS2とS1は、音声符号化のフレーム更新の際に、前のフレームでのフラグS1、S0の内容をシフトしたものである。S0フラグの内容は、状態フラグ更新部351において対応するバッファ34からの入力フレームの誤り検出情報と、S1フラグ(これは、前フレームではS0フラグに相当する)の内容に従って決定される。

【0016】図4にS0の更新表を示す。図4において、バッファ34の入力=0のとき、その入力フレームは、誤り訂正が完全にできたことを示し、バッファ34の入力=1のとき、誤り訂正ができなかった、つまり、誤りが検出されたフレームであることを意味する。S1が0のとき、入力フレームの誤りが検出されなければ、S0=0とし、誤りが検出されれば、S0=1とする。S1が1からあらかじめ与えられた数M-1までの場合には、入力フレームの誤りが検出されなければS0=0、誤りが検出されれば、S0=S1+1として、対応するフレームまで、誤りフレームの連続して続いたフレーム数をフラグの内容として保持する。S1がMの場合には、入力フレームで誤りが検出されなければ、S0=M+1とし、誤りが検出されれば、S0=Mとする。ここで、Mは、連続するフレームにバースト性の誤りが加わっていると考えられる長さ、例えば6、とする。S1がM+1の場合には、入力フレームで誤りが検出されなければ、S0=0、誤りが検出されれば、S0=Mとする。従って、S0=M+1の場合には、対応フレームの誤り検出がない場合であるが、バースト誤りが続いた後であるので、0とはせず、さらにその次のフレームで、誤りが検出されなくなると0にする。

【0017】このような状態フラグの遷移に関しては、従来の技術で述べたM. J. McLaughlinの文献(電子情報通信学会無線通信システム研究会、RCS90-27, pp. 41-45, 1990年)において、現在フレームを入力として、S0を現在フレームのフラグとして、過去のフラグとともに用いているが、本発明では、現在フレームがS1、将来のフレームをS0として、これから復号しようとするフレームの次に来るフレームのフラグを用いる点が大きく異なる。

【0018】次に、こうして得られた誤り管理制御部35の出力S(S2, S1, S0)を用いて、図2のパラメータ復号部32で過去、現在、将来フレームの誤り状態を考慮した各音声符号化情報のパラメータ復号を行なう。パラメータ復号部32での誤り状態フラグに依存した処理手順のフロー図の例を図5に示す。まず、図5のステップ101において、現在フレームの誤り状態フラグS1が0であるかどうかを判断し、0ならば、ステップ111の処理1を実行

声符号化情報に基づいて復号したパラメータを基本とし、 $S_0, S_2$  を見て、 $S_0$  または  $S_2$  が 0 でない場合には、連続的にパラメータが接続するように修正する。

【0019】次に、ステップ102において、 $S_1$  が 0 でなく、フラグが 1 から  $M$  のいずれかの場合には、現在フレームにおいて、誤り検出されたことを示している。従って、この場合は、現在フレームの音声符号化情報には、誤りが含まれている。そこで、過去フレーム及び将来フレームの情報を使うことになる。ステップ103において、 $S_0=0$  または  $M+1$  ならば将来フレームが正しいので、ステップ112の処理2として、過去フレームで復号に使われたパラメータと将来フレームの復号パラメータから現在フレームのパラメータを両側から内挿補間する。その際、 $S_1$  のフラグの誤り継続の度合いにより、内挿の重みを変えることもできる。例えば、 $S_1$  が  $M$  に近ければ、過去フレームのパラメータは、バースト誤りが継続して、次に述べるステップ113の処理3の外挿的な手法によって得られたパラメータであるので、過去のパラメータの重みは小さく、将来フレームの重みを大きくすることにより、より歪の小さい現在フレームのパラメータを得ることができる。ステップ103において、 $S_0 \neq 0$  かつ  $S_0 \neq M+1$  の場合、将来フレームには誤りが検出されている。その場合、ステップ113の処理3では、過去フレームで復号に使われたパラメータの現在フレームへの外挿による補間処理を行なう。その際、 $S_1$  フラグの誤り継続の度合いにより、外挿結果への重みを変えることもできる。

【0020】ステップ102において、 $N_0$  の場合には、 $S_1 = M+1$  であり、これは、過去のフレームに  $M$  フレーム以上のバースト誤りがあった後に現在フレームが正しいことを示している。従って、ステップ114の処理4では、現在フレームの音声符号化情報に基づいて復号したパラメータを基本とし、さらに、 $S_2=M$  であるから、過去フレームにバースト誤りがあったこと、及び、 $S_0$  が 0 または  $M$  になるので、その値によって、連続的にパラメータが接続するように修正する。

【0021】次に、実際のパラメータについて、図5の処理1～処理4の例について述べる。CELP符号化における各フレームの音声符号化情報は、図1に示したように、線形予測パラメータ符号A、パワパラメータ符号P、ピッチ周期符号Lと雑音符号帳の雑音符号ベクトル符号C、音源のゲインパラメータ符号Gからなる。まず、線形予測パラメータ符号Aについての実施例を示す。線形予測パラメータとしては、補間の際には、線形補間特性の良い線スペクトル対 (LSP : Line Spectrum Pair) を用いる。線形予測パラメータは、今1フレームを20msとすれば、実際には、5ms～10ms程度のサブフレーム単位毎にパラメータを必要とする。その場合、前フレーム (過去フレーム) または、後フレーム (将来フ

レームのパラメータを求める。すると、前 (または、後) フレームに符号誤りがある場合には、誤ったパラメータと直接補間すると、大きな歪が生じる。また、現在フレームのパラメータが誤っている場合でもすぐ前後のパラメータが誤りのない状態あれば、前後フレームから、現在フレームを補間することができる。

【0022】過去 (時刻  $n-1$  とする)、現在 (時刻  $n$ )、将来 (時刻  $n+1$ ) フレームでの誤り訂正直後で補間をしないLSPパラメータをそれぞれ、 $\omega(n-1)$ 、 $\omega(n)$ 、 $\omega(n+1)$  とし、誤り状態フラグを考慮して補間したあとの時刻  $n-1$ 、 $n$  に対応するパラメータを  $\Omega(n-1)$ 、 $\Omega(n)$  で表す。また、 $a \leq l \leq b$  の範囲の  $l$  に対して、時刻  $a$  で  $X_a$ 、時刻  $b$  で  $X_b$  のパラメータであるときの時刻  $l$  での内挿パラメータ  $\Omega(l)$  を、

$$\Omega(l) = f\_A(l; (a, X_a), (b, X_b))$$

で定義する。関数  $f\_A$  は、2点  $(a, X_a)$ 、 $(b, X_b)$  間を補間する関数で、時刻  $l$  の補間値を出力するものである。例えば、直線補間でも良い。

【0023】図5ステップ111の処理1では、現在フレームの誤り状態フラグ  $S_1=0$  であり、 $\omega(n)$  には誤りがないので、 $S_2, S_0$  の誤り状態に応じて次のようにすれば良い。現在フレーム内の  $l$  において、 $n-1 \leq l \leq n$  では、 $S_2=0$  ならば、 $\Omega(l) = f\_A(l; (n-1, \Omega(n-1)), (n, \omega(n)))$ 、 $S_2=1, \dots, M$  ならば、 $\Omega(l) = f\_A(l; (n-1, \omega(n)), (n, \omega(n)))$

とする。また、 $n \leq l \leq n+1$  のとき、

$$S_0=0 \text{ ならば、} \Omega(l) = f\_A(l; (n, \omega(n)), (n+1, \omega(n+1)))$$

$$S_0=1, \dots, M \text{ ならば、} \Omega(l) = f\_A(l; (n, \omega(n)), (n+1, \omega(n)))$$

とする。

【0024】図5のステップ112の処理2では、現在フレームパラメータに誤りがあるので  $\omega(n)$  は使用せず、 $S_0=0$  または  $M+1$  なので、 $\Omega(n-1)$ 、 $\omega(n+1)$  を用いて、現フレーム内の時刻  $l$  のパラメータを、

$$\Omega(l) = f\_A(l; (n-1, \Omega(n-1)), (n+1, \omega(n+1)))$$

により補間する。ステップ113の処理3では、 $S_0=1, \dots, M$  なので、過去のパラメータを用いて、 $\Omega(l) = f\_A(l; (n-1, \Omega(n-1)), (n+1, \Omega(n-1)))$  により、現フレーム内の時刻  $l$  のパラメータを補間する。この場合、 $S_2, S_1, S_0$  の値に応じて  $\Omega(l)$  のスペクトルのバンド幅を拡張するような関数を用いて出力のLSPパラメータの共振特性を弱めるようにして、歪を防ぐような補間をしても良い。

【0025】図5のステップ114の処理4では、 $\omega(n)$  は使えるが、前フレームがバースト誤りなので、 $\Omega(n-1)$  を使うか、または  $\Omega_0$  として、平坦スペクトルとなるようなパラメータのどちらかを用いる。これを  $\Omega'$  と

き、

$$\Omega(t) = f\_A(t; (n-1, \Omega'), (n, \omega(n))),$$

$n \leq t \leq n+1$  では、

$$S0=0 \text{ ならば、 } \Omega(t) = f\_A(t; (n, \omega(n)), (n+1, \omega(n+1))),$$

$$S0=M \text{ ならば、 } \Omega(t) = f\_A(t; (n, \omega(n)), (n+1, \omega(n)))$$

とする。

【0026】次に、パワパラメータ符号の図5の各処理について考える。基本的には、線形予測パラメータの補間の場合と同様である。パワの場合もサブフレーム単位のパワを計算する場合について示す。過去(時刻 $n-1$ とする)、現在(時刻 $n$ )、将来(時刻 $n+1$ )フレームでの誤り訂正直後で補間をしないパワパラメータをそれぞれ、 $p(n-1)$ 、 $p(n)$ 、 $p(n+1)$ とし、誤り状態フラグを考慮して補間したあとの時刻 $n-1$ 、 $n$ に対応するパラメータを $P(n-1)$ 、 $P(n)$ とする。

【0027】時刻 $a$ で $Xa$ 、時刻 $b$ で $Xb$ であるときの時刻 $t$ でのパワの内挿パラメータ $P(t)$ を、

$$P(t) = f\_P(t; (a, Xa), (b, Xb))$$

で定義する。関数 $f\_P$ は、2点 $(a, Xa)$ 、 $(b, Xb)$ 間の直線補間、または、 $Xa$ 、 $Xb$ の対数表現の直線間補といった関数でも良い。

【0028】図5のステップ111の処理1では、 $S1=0$ なので、 $p(n)$ には、誤りがないので、現フレーム内の $t$ において、 $n-1 \leq t \leq n$ では、

$$S2=0 \text{ ならば、 } P(t) = f\_P(t; (n-1, P(n-1)), (n, p(n))),$$

$$S2=1, \dots, M \text{ ならば、 } P(t) = f\_P(t; (n-1, p(n)), (n, p(n)))$$

とする。そして、結果の $P(t)$ が、 $P(n-1)$ よりも一定の定数倍(例えば $\alpha \geq 1$ )よりも大きい場合には、 $\alpha' P(t)$ 、( $0 \leq \alpha' \leq 1$ )として、パワを小さく制限することにより、急激なパワ変化による歪を聞こえないようにする。また、 $n \leq t \leq n+1$ では、

$$S0=0 \text{ ならば、 } P(t) = f\_P(t; (n, p(n)), (n+1, p(n+1))),$$

$$S0=1, \dots, M \text{ ならば、 } P(t) = f\_P(t; (n, p(n)), (n+1, p(n)))$$

とする。

【0029】図5のステップ112の処理2では、現在フレームパラメータに誤りがあるので、 $p(n)$ は使用せずに、 $S0=0$ または $M+1$ なので、 $P(n-1)$ 、 $p(n+1)$ を用いて、現フレーム内の時刻 $t$ のパラメータを、

$$P(t) = f\_P(t; (n-1, P(n-1)), (n+1, p(n+1)))$$

により補間する。ここで、一定の閾値(例えば $\alpha \geq 1$ )に対して、 $P(t) > \alpha P(n-1)$ であり、パワ変化が大きすぎる場合には、 $\alpha' P(t)$ 、( $0 \leq \alpha' \leq 1$ )のようにパワを置換し、パワを小さく制限することにより、歪が聞こえないようにする。 $\alpha'$ は、 $S1$ のバースト誤りの継続数

【0030】ステップ113の処理3では、 $S0=1, \dots, M$ なので、過去のパラメータを用いて、

$$P(t) = f\_P(t; (n-1, P(n-1)), (n+1, P(n-1)))$$

とし、さらに、誤りフレームの音量を弱めて $P(t)$ を $\alpha' P(t)$ 、( $0 \leq \alpha' \leq 1$ )とする。 $\alpha'$ は、 $S1$ のバースト誤りの継続数によって変更しても良い。例えば、 $S1=M$ ならば、現在フレームまでに、 $M$ フレーム以上の大きなバースト誤りが生じていることを意味するので、 $\alpha' = 0$ として無音区間とし、歪みのある音声を出ししないようにする。

【0031】図5のステップ114の処理4では、 $p(n)$ は使えるが、前フレームがバースト誤りフレームである。そこで、現フレーム内の $t$ において、 $n-1 \leq t \leq n$ のとき、

$$P(t) = f\_P(t; (n-1, P(n-1)), (n, p(n)))$$

そして、 $P(t)$ を $\alpha' P(t)$  ( $0 \leq \alpha' \leq 1$ )に置き換える。 $n \leq t \leq n+1$ では、

$$S0=0 \text{ ならば、 } P(t) = f\_P(t; (n, p(n)), (n+1, p(n+1))),$$

$$S0=M \text{ ならば、 } P(t) = f\_P(t; (n, p(n)), (n+1, p(n)))$$

とする。 $S0=M$ の場合は、さらに、 $P(t)$ を $\alpha' P(t)$ 、( $0 \leq \alpha' \leq 1$ )とする。

【0032】次に、ピッチ周期符号 $L$ のパラメータ復号の例を示す。ピッチ周期符号 $L$ は、図1において、適応符号帳10のピッチ遅延を示す遅延値である。これは、1フレームを $k$ 個(例えば2個)のサブフレームに分け、その個数だけ $L$ の値をピッチ情報として伝送する。ピッチ遅延は、CELPではピッチ周期だけでなく2倍周期、3倍周期といったとびとびの値をとることもある。従って、本実施例では、 $k=2$ とした場合について、前フレームと後のフレームとのピッチ周期の値を現フレームが誤ったときに置換するという手法について示す。もちろん、ピッチ周期の変動が小さく、前後フレームのピッチ周期を補間可能な場合には、補間によって求めてもよい。

【0033】過去(時刻 $n-1$ )、現在(時刻 $n$ )、将来(時刻 $n+1$ )フレームでの誤り訂正直後ピッチ遅延をそれぞれ、 $1k(n-1)$ 、 $1k(n)$ 、 $1k(n+1)$ 、( $k=1, 2$ )とし、誤り状態フラグを考慮したあとの時刻 $n-1$ 、 $n$ に対応するパラメータを $Lk(n-1)$ 、 $Lk(n)$ 、( $k=1, 2$ )とする。図5のステップ111の処理1では、 $S1=0$ なので、 $1k(n)$ に誤りがないので、 $Lk(n) = 1k(n)$ 、( $k=1, 2$ )とする。

【0034】図5のステップ112の処理2では、現在フレームのパラメータに誤りがあるので、 $1k(n)$ は使用せずに、

$$L1(n) = L2(n-1), L2(n) = L1(n+1)$$

とする。ステップ113の処理3では、 $S0=1, \dots, M$ なので



$L_k(n) = L_2(n-1), (k=1, 2)$

とする。

【0035】ステップ114 の処理4では、現在フレームが正しいので、処理1と同じく、

$L_k(n) = l_k(n), (k=1, 2)$

とする。次に、音源ゲインパラメータGの復号例を示す。Gは、図1において、適応符号帳10と雑音符号帳11で選ばれた各音源信号に乗ずるゲイン $g_1, g_2$ の組を総称している。このGに関しては、ピッチ遅延と同様にパラメータ復号を行なう。過去(時刻 $n-1$ )、現在(時刻 $n$ )、将来(時刻 $n+1$ )フレームでの誤り訂正直後音源ゲインをそれぞれ、 $\sigma_k(n-1), \sigma_k(n), \sigma_k(n+1), (k=1, 2)$ とし、誤り状態フラグを考慮したあとの時刻 $n-1, n$ に対応するパラメータを $G_k(n-1), G_k(n), (k=1, 2)$ とする。

【0036】図5のステップ111 の処理1では、 $S1=0$ であり $\sigma_k(n)$ に誤りがないので、

$G_k(n) = \sigma_k(n), (k=1, 2)$

とする。図5のステップ112の処理2では、現在フレームパラメータに誤りがあるので、 $\sigma_k(n)$ は使用せずに、 $G_1(n) = G_2(n-1), G_2(n) = \sigma_1(n+1)$

とする。

【0037】ステップ113 の処理3では、 $S0=1, \dots, M$ なので過去のパラメータを用いて、

$$H(z) = \frac{(1 - \eta z^{-1})A(z/\gamma_1)}{((1 - \delta B(z))A(z/\gamma_2))}$$

$$A(z) = \sum_{i=0}^n \alpha_i z^{-i}$$

$$B(z) = \sum_{i=-1}^1 \beta_i z^{-\tau-i}$$

【0041】ここで、 $\alpha_i$ は、各フレームの線形予測係数であり、 $\beta_i$ は長期予測係数、 $\tau$ はピッチ遅延である。これらは、伝送パラメータあるいは、復号音声から適応的に決定される。

$A(z/\gamma_1)/A(z/\gamma_2)$

でホルマント強調を行い、 $1/(1 - \delta B(z))$ でピッチの強調を、さらに、

【0042】

【数2】

$$1 - \eta z^{-1}$$

【0043】でスペクトル傾斜補正の効果をもたせている。本実施例では、 $\eta, \delta, \gamma_1, \gamma_2$ を誤り状態フラ

$G_k(n) = G_2(n-1), (k=1, 2)$

とする。ステップ114 の処理4では、現在フレームが正しいので、処理1と同じく、

$G_k(n) = \sigma_k(n), (k=1, 2)$

とする。

【0038】雑音符号帳の雑音ベクトル符号Cの復号に関しては、フレーム間の相関は、大きくないので、本実施例では、誤り状態フラグに関係なく現在フレームに送られた符号をそのまま再生に使用する。こうして、図2のパラメータ復号部32で、過去、現在、将来の誤り状態フラグに基づき、音声符号化情報のパラメータが復号・修復される。その結果が、次の音声復号器33に渡され、音声再合成される。

【0039】音声復号器33においても、誤り状態管理部35からの過去、現在、将来のフレーム誤り状態フラグ $S(=S2, S1, S0)$ を用いて、復号音声の再生音声を制御することにより伝送誤りによる歪みを小さくできる。すなわち、図1のポストフィルタ13において、 $S$ を用いてポストフィルタの係数を制御して聴覚的な歪みを抑える。通常、ポストフィルタは、次のような $h(z)$ によって構成されている。

【0040】

【数1】

的に変化させる。 $S2, S1, S0$ がいずれも0または $M+1$ の場合には、通常値を使用するが、 $S2, S1, S0$ のいずれかが0と $M+1$ 以外で誤りフレームである場合には、適応符号帳や、雑音符号ベクトル符号に誤りが含まれている可能性が高いので、 $\eta$ と $\delta$ を小さくして、高域のスペクトル傾斜と音源のピッチ強調を弱め、 $\gamma_1$ と $\gamma_2$ を調整してホルマント強調の度合いを強めることにより、出力音声の雑音感を抑えることができる。ここで、 $S0$ の将来フレームまでの誤り状態フラグを用いることにより、 $S2, S1$ だけの場合よりも $\eta, \delta, \gamma_1, \gamma_2$ の切替えを連続的に行うことが可能となり、フレーム間での切替えによる音質変化による劣化を抑えることができる。

【0044】

音声符号化ビットに伝送路の符号誤りが残留した場合に、復号音声に重畳する耳障りな歪みを押さえることが可能となる。

【図面の簡単な説明】

【図 1】 CELP 音声復号器のブロック図である。

【図 2】 本発明の実施例における音声復号処理のブロック図である。

【図 3】 図 2 の誤り状態管理部 35 の拡大図である。

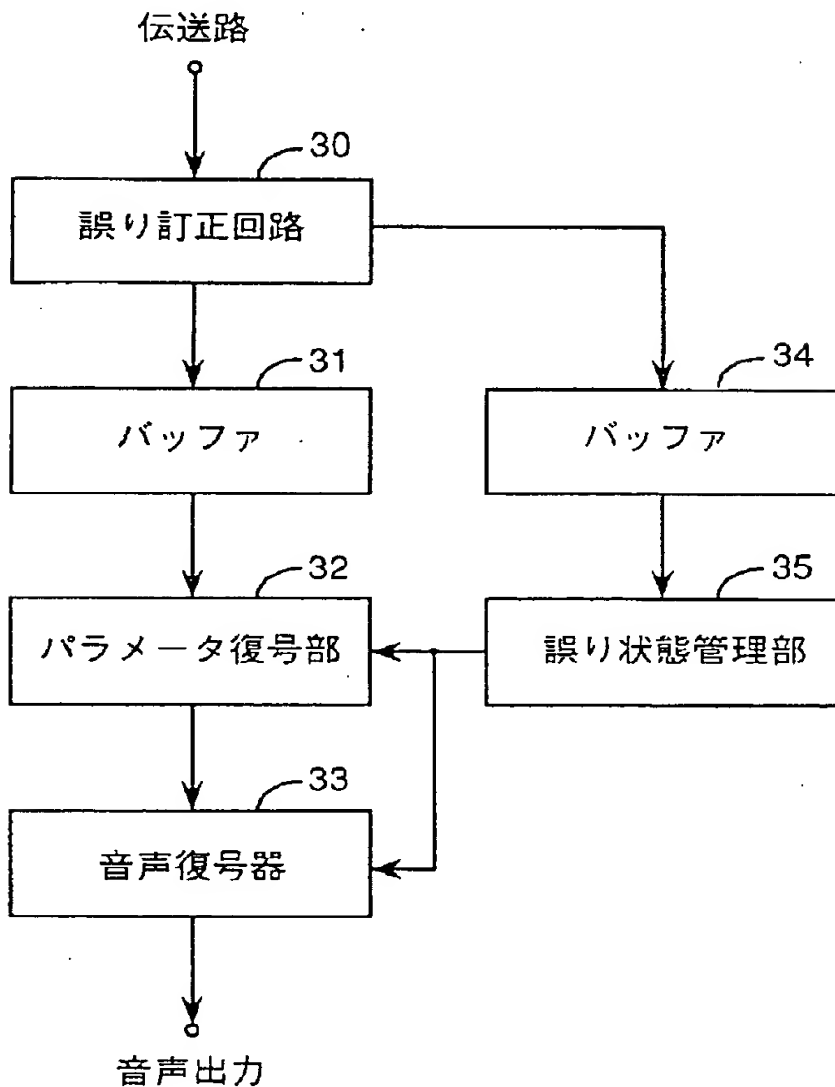
【図 4】 図 3 の状態フラグ更新部 351 における、S0 フラグの更新表である。

【図 5】 誤り状態フラグに応じた処理フローの概略図である。

【符号の説明】

- 1 0 は適応符号帳  
1 1 は雑音符号帳  
1 2 は線形予測合成フィルタ  
1 3 はポストフィルタ  
2 1, 2 2, 2 4 は乗算器  
2 3 は加算器  
3 0 は誤り訂正回路  
3 1, 3 4 はバッファ  
3 2 はパラメータ復号部  
3 3 は音声復号器  
3 5 は誤り状態管理部

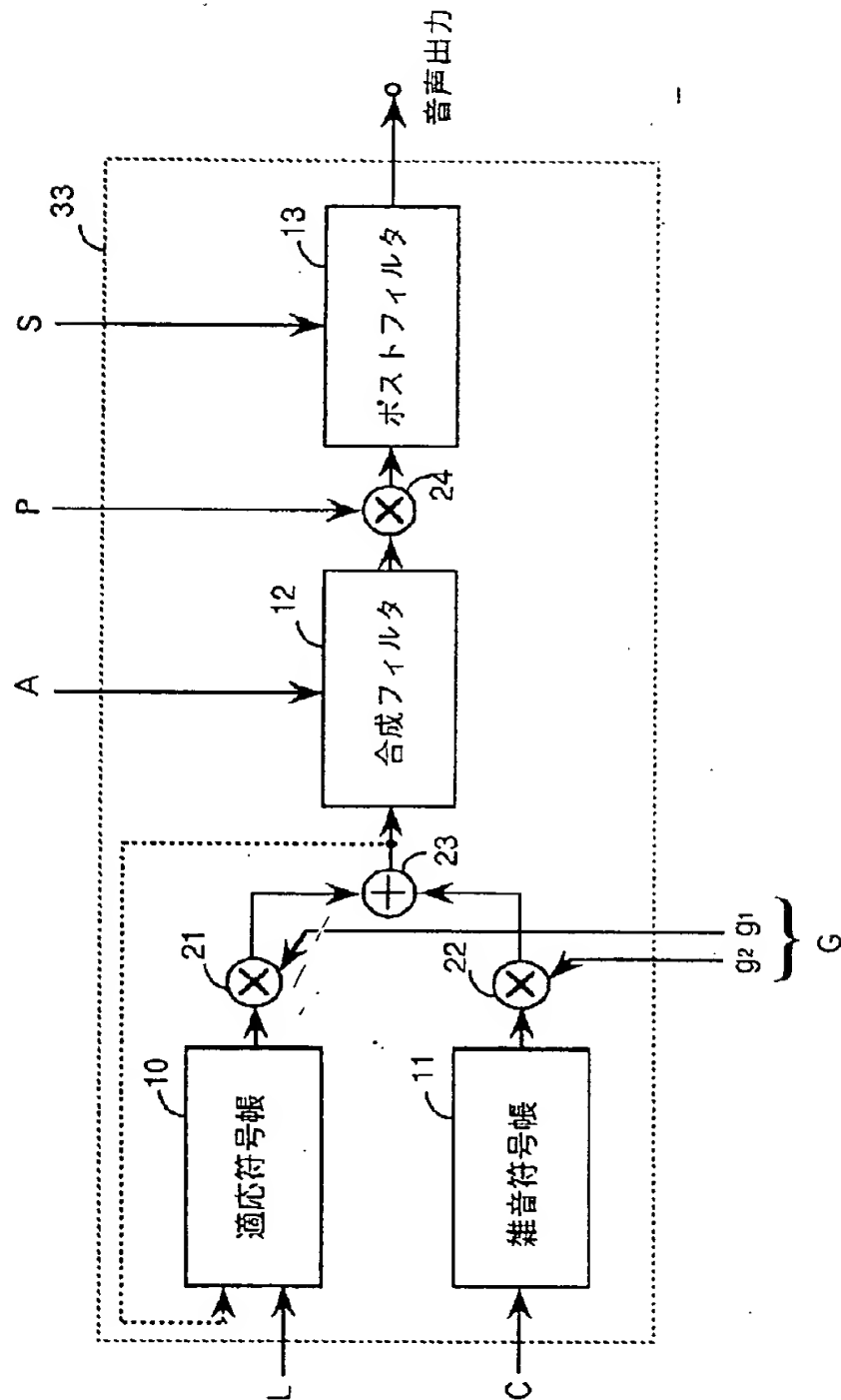
【図 2】



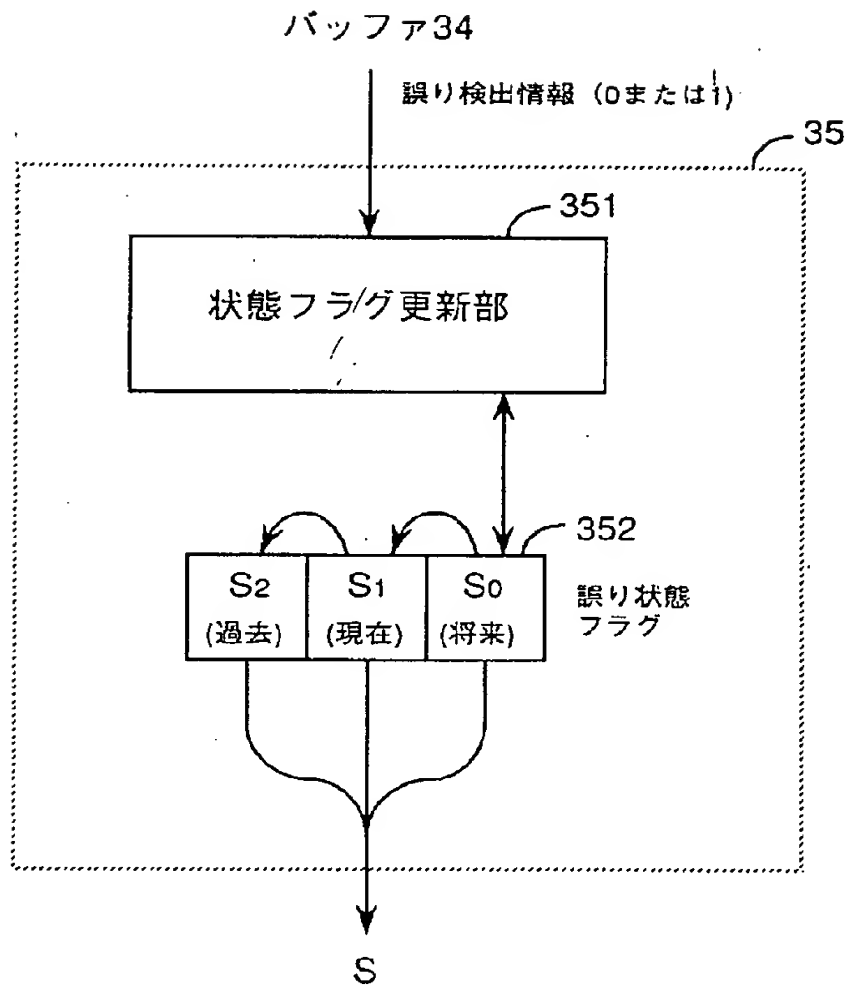
【図 4】

Soの更新		
新S <sub>i</sub> (旧S <sub>0</sub> )	バッファ34 入力	新S <sub>0</sub>
0	0	0
1	0	0
2	0	0
⋮	⋮	⋮
M-1	0	0
M	0	M+1
M+1	0	0
0	1	1
1	1	2
2	1	3
⋮	⋮	⋮
M-1	1	M
M	1	M
M+1	1	M

【図1】



【図3】



【図5】

